MULTIMEDIA UNIVERSITY

STUDENT ID NO

MULTIMEDIA UNIVERSITY

FINAL EXAMINATION

TRIMESTER 3, 2016/2017

ECP2216 – MICROCONTROLLER AND MICROPROCESSOR SYSTEMS

(All sections / Groups)

31 MAY 2017
2.30 p.m – 4.30 p.m
(2 Hours)

INSTRUCTIONS TO STUDENTS

1. This Question paper consists of 10 pages with 5 Questions only.

2. Attempt **ALL FIVE COMPULSORY** questions. All questions carry equal marks and the distribution of the marks for each question is given.

3. Please write all your answers in the Answer Booklet provided.

Question 1

(a) Determine the range of a 10-bit two's complement number.

[2 marks]

(b) A memory IC is specified as 1M*4Bytes. Evaluate the number of address lines and data lines required for this memory IC.

[3 marks]

(c) Describe THREE perspectives to differentiate microcontrollers from microprocessors.

[6 marks]

(d) (i) Highlight THREE advantages of using multi-core processor system.

[3 marks]

(ii) State six functional units in 80386DX Intel 32-bit microprocessor.

[6 marks]

Continued ...

Question 2

(a) Consider this instruction: **MOV 22H, # A4H.** Identify which bit addresses are set to 1 after the execution of this instruction.

[3 marks]

(b) After the execution of the following instruction sequence, determine the contents of Program Status Word (PSW) and Accumulator (A) upon system RESET.

MOV A, #0D7H
MOV R3, #49H
ADD A, R3

(Hints: Express your answer in hexadecimal format.)

[4 marks]

(c) Design an 8051 microcontroller based system that can address 8Kbytes of RAM which should occupy the first portion of the memory space followed by 8Kbytes of ROM. Assume that the available memory ICs are 4Kbytes ROM and 4Kbytes RAM.
(*Hints: Draw and label the system configuration showing the 8051 signal lines to be used for data, address and control buses.*)

[13 marks]

Continued ...

Question 3

(a) Upon system RESET, determine the contents of the accumulator (ACC) and PSW register after the execution of **EACH** instruction in the following MCS-51 assembly language program. *Express all the values in hexadecimal format.*

```
ORG  0000H
MOV  A, #00100101B
INC  A
ADD  A, #0FFH
SUBB A, #22
END
```

[6 marks]

(b) State any **THREE** available addressing modes for MCS-51 program arithmetic instructions.

[3 marks]

(c) Consider the following MCS-51 assembly language subroutine which is called **DELAYRoutine**:

```
DELAYRoutine:  MOV  R0, #100
LOOP:          NOP
               DJNZ  R0, LOOP
               RET
```

(i) Calculate the total execution time of **DELAYRoutine**. (*Assume a 12 MHz crystal frequency is used.*)

[3 marks]

(ii) Assume the first instruction of the **DELAYRoutine** is addressed at ***0010H***. Convert the instruction "**DJNZ R0, LOOP**" into the corresponding machine code.

[5 marks]

(iii) Based on **DELAYRoutine**, add instructions to increase the total execution time to 3 milliseconds.

[3 marks]

Continued ...

Question 4

(a) Assume 11.0592MHz crystal frequency, 8-bit data, 1 stop bit, no parity and operation at 4800 baud rate generated by Timer 1. Write a MCS-51 assembly language sequence to show the initialization of SCON, TMOD and TH1 registers. *You may need to show the calculation steps to obtain the value of TH1.*

[5 marks]

(b) Describe the procedures of an 8051 microcontroller needs to go through upon occurrence of an interrupt.

[5 marks]

(c) Consider an 8051 microcontroller system interfaces with two switches that are normally HIGH. First switch is connected to INT1 pin and second switch is connected to INT0 pin. Write a MCS-51 assembly language program to perform the following tasks.

1. When switch connected to INT0 goes LOW, timer 0 interrupt will generate a pulse width of 0.2ms at P2.0.

2. When switch connected to INT1 goes LOW, timer 1 interrupt will generate a pulse width of 0.3ms at P0.3.

[10 marks]

Continued ...

Question 5

(a) An 8051 microcontroller interfaces to a common cathode-type seven-segment LED display device on Port 1 and one press button on P3.0 is shown in Figure 5(a).

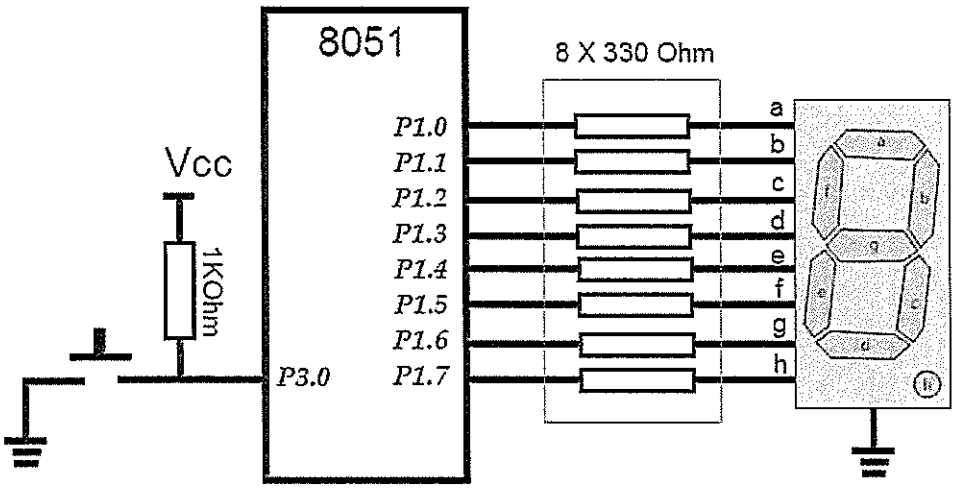

Figure 5(a)

(i) Complete Table 5(a) to express the bit patterns for each character to decode the seven-segment LED display.

Table 5(a)

	P1.7	*P1.6*	*P1.5*	*P1.4*	*P1.3*	*P1.2*	*P1.1*	*P1.0*
Character	h	g	f	e	d	c	b	a
9								
A								

[2 marks]

(ii) Write a MCS-51 assembly language program that will wait for the button press on P3.0. Once the button is pressed and hold, seven-segment LED display will repeatedly display the characters in sequence starting from 9 and A. Time duration for each character to be displayed is 1 second. The display will be stopped only if the button is released. *Assume 1 second delay subroutine DELAY1S is available.*

[8 marks]

Continued ...

(b) Figure 5(b) illustrates an 8051 microcontroller based ***Automated Muffin Powder Mixer Machine***:

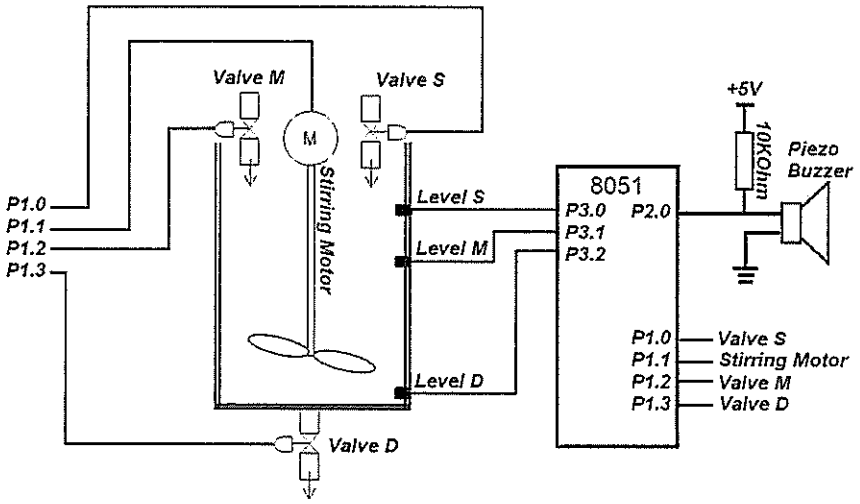

Figure 5(b)

The machine performs the following process:

1. The chamber is first filled with unsweetened muffin flour through a solenoid Valve M.
2. When the unsweetened muffin flour reaches Level M, Valve M is closed and the chamber is now filled with sweetener powder through Valve S.
3. When the mixture in the chamber reaches Level S, Valve S is closed.
4. The stirring motor starts the mixing process that last for approximately 1 minutes.
5. After that, the drainage Valve D opens to dispense the mixture.
6. When the mixture reaches Level D, Valve D is closed and the piezo buzzer will sound for approximately 1 minute to indicate the completion.
7. The whole process is repeated from Step 1 again.

The chamber has three level sensors that send signals to input lines ***P3.0*** to ***P3.2***. A logical **HIGH** from the sensor indicates that the level has been reached. The output lines ***P1.0***, ***P1.2***, and ***P1.3*** provide signals to the solenoid valves. A logical **HIGH** from the lines will open the corresponding valve. The output lines ***P1.1*** and ***P2.0*** provide signals to the mixer motor and buzzer respectively which are both activated by a logical **HIGH**.

Write a MCS-51 assembly language program to carry out the process. Assume 12MHz crystal frequency is used

[10 marks]

End of Page

APPENDIX

Special Function Register Formats

Interrupt Enable (IE)

Bit Addr.	AFH	-	-	ACH	ABH	AAH	A9H	A8H
Name	EA	-	-	ES	ET1	EX1	ET0	EX0

BIT	SYMBOL	FUNCTION (Enable=1, Disable=0)
IE.7	EA	Global enable/disable. EA = 1, each individual source is enabled/disabled by setting/clearing its enable bit. EA = 0, disable all interrupts.
IE.6	-	Undefined
IE.5	-	Not implemented in 8051. ET2 for 8052.
IE.4	ES	Serial port interrupt enable bit.
IE.3	ET1	Timer 1 interrupt enable bit.
IE.2	EX1	External interrupt enable bit.
IE.1	ET0	Timer 0 interrupt enable bit.
IE.0	EX0	External interrupt enable bit.

Interrupt Priority (IP)

Bit Addr.	-	-	-	BCH	BBH	BAH	B9H	B8H
Name	-	-	-	PS	PT1	PX1	PT0	PX0

BIT	SYMBOL	FUNCTION (Enable=1, Disable=0)
IP.7	-	Undefined
IP.6	-	Undefined
IP.5	-	Not implemented in 8051. PT2 for 8052.
IP.4	PS	Serial port interrupt priority bit.
IP.3	PT1	Timer 1 interrupt priority bit.
IP.2	PX1	External interrupt priority bit.
IP.1	PT0	Timer 0 interrupt priority bit.
IP.0	PX0	External interrupt priority bit.

Interrupt Vectors

Interrupt Source	Flag	Vector Address
System Reset	RST	0000H
External 0	IE0	0003H
Timer 0	TF0	000BH
External 1	IE1	0013H
Timer 1	TF1	001BH
Serial Port	RI & TI	0023H
Timer 2 (8052)	TF2 or EXF2	002BH

Program Status Word (PSW)

Bit Addr.	D7H	D6H	D5H	D4H	D3H	D2H	-	D0H
Name	CY	AC	F0	RS1	RS0	0V	-	P

Serial Control (SCON)

Bit Addr.	9FH	9EH	9DH	9CH	9BH	9AH	99H	98H
Name	SM0	SM1	SM2	REN	TB8	RB8	TI	RI

BIT	SYMBOL	FUNCTION
SCON.7	SM0	Serial port mode bit 0 (see Table A.1).
SCON.6	SM1	Serial port mode bit 1 (see Table A.1).
SCON.5	SM2	Serial port mode bit 2; enables multiprocessor communications in modes 2 and 3; RI will not be activated if received 9th bit is 0. In mode 1, if SM2 = 1, then RI will be activated only if a valid stop bit was received. In mode 0, SM2 should be 0.
SCON.4	REN	Receiver enable; must be set to receive characters.
SCON.3	TB8	Transmit bit 8; 9th bit transmitted in modes 2 and 3; set/cleared by software.
SCON.2	RB8	Receive bit 8; 9th bit received.
SCON.1	TI	Transmit interrupt flag; set at end of character transmission; cleared by software.
SCON.0	RI	Receive interrupt flag; set at end of character reception; cleared by software.

Table A.1 The 8051 Serial Port Mode Selection

SM0	SM1	Mode	Description	Baud Rate
0	0	0	Shift register	Fixed
0	1	1	8-bit UART	Variable
1	0	2	9-bit UART	Fixed
1	1	3	9-bit UART	Variable

Timer Control (TCON)

Bit Addr.	8FH	8EH	8DH	8CH	8BH	8AH	89H	88H
Name	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0

BIT	SYMBOL	FUNCTION
TCON.7	TF1	Timer-1 overflow flag. Set by hardware on overflow. Cleared by hardware when processor vectors to interrupt routine. Must be cleared by software when not involve interrupt
TCON.6	TR1	Timer-1 run control bit. Set/cleared by software to turn timer/counter on/off.
TCON.5	TF0	Timer-0 overflow flag. Do the same function as TF1 but for Timer-0.
TCON.4	TR0	Timer-0 run control bit. Do the same function as TR1 but for Timer-0.
TCON.3	IE1	External interrupt-1 edge flag. Set by hardware when interrupt-1 falling edge is detected. Cleared by hardware when interrupt is processed.
TCON.2	IT1	Interrupt-1 Type control bit. Set/cleared by software to specify falling edge/low level triggered external interrupts.
TCON.1	IE0	External interrupt-1 edge flag. Do the same function as IE1 but for external interrupt-0.
TCON.0	IT0	Interrupt-0 Type control bit. Do the same function as IT1 but for external interrupt-0.

Timer Mode (TMOD)

Bit	7	6	5	4	3	2	1	0
Name	GATE	C/T	M1	M0	GATE	C/T	M1	M0

BIT	SYMBOL	FUNCTION
TMOD.7	GATE1	When this bit is set the timer will only run when INT1 (P3.3) is high (hardware control). When this bit is cleared the timer will run regardless of the state of INT1 (software control).
TMOD.6	C/T1	Timer / Counter select bit. C / $\overline{T}$ = 0 → Timer operation. C / $\overline{T}$ = 1 → Counter operation.
TMOD.5	M1	Mode selection bits (see Table A.2). [for timer 1]
TMOD.4	M0	Mode selection bits (see Table A.2). [for timer 1]
TMOD.3	GATE0	Exactly the same function as GATE1 but for Timer0
TMOD.2	C/T0	Exactly the same function as C/T1 but for Timer0
TMOD.1	M1	Mode selection bits (see Table A.2). [for timer 0]
TMOD.0	M0	Mode selection bits (see Table A.2). [for timer 0]

Table A.2 Timer Mode Selection

M1	M0	Timer Mode	Description of Mode
0	0	0	13-bit Timer
0	1	1	16-bit Timer
1	0	2	8-bit auto-reload
1	1	3	Split timer mode

MCS-51 Opcode Map